Many-Core Acceleration of Common Graph Programming Frameworks

NOTE: The Solicitations and topics listed on this site are copies from the various SBIR agency solicitations and are not necessarily the latest and most up-to-date. For this reason, you should use the agency link listed below which will take you directly to the appropriate agency server where you can read the official version of this solicitation and download the appropriate forms and rules.

The official link for this solicitation is:
http://www.acq.osd.mil/osbp/sbir/solicitations/sbir20152/index.shtml

Agency:
Department of Defense

Release Date:
April 24, 2015
Branch:
n/a

Open Date:
April 24, 2015
Program / Phase / Year:
SBIR / Phase I / 2015

Application Due Date:
June 24, 2015

Solicitation:
 DoD 2015.2 SBIR Solicitation

Close Date:
June 24, 2015
Topic Number:
SB152-004

Description:
  Today there is a DoD need for graph analytics capabilities, which are critical for a large range of application domains with a vital impact on both national security and the national economy, including, among others: counter-terrorism; fraud detection; drug discovery; cyber-security; social media; logistics and supply chains; e-commerce, etc. Widely used graph development frameworks have enabled online (but not real-time) graph analytics for broad classes of problems at a modest data scales and support only offline analytics for very large data scales. The Facebook graph today has over 1 Trillion edges. A single iteration of a graph traversal takes up to 3 minutes using Apache Giraph on 200 commodity CPU servers. A full breadth first traversal of the graph could take nearly 20 minutes, and algorithms that relax to a solution can require 50-100 iterations, implying that it could take several hours to compute the Page Rank of the Facebook graph. Bringing analytics within these graph programming frameworks into real-time on large graphs requires that they be able to leverage the computing advances in multi-core platforms. However, scalable, data-parallel graph analytics on many-core hardware is a fundamentally hard problem that goes well beyond the current state of the art. Graph data models and algorithms are used for network structured data, when the data are poorly structured, or when complex relationships must be drawn from multiple data sets and analyzed together. Graph operations are inherently non-local and, for many real-world data sets, that non-locality is aggravated by extreme data skew. Graph analytics are data intensive rather than compute intensive which means that memory and network bandwidth are the bottlenecks for graph

processing. Overall, current solutions applied to scaling graph frameworks such as Tinkerpop and Graphlab do not have all of the desired attributes integrated, specifically 1) Solutions based on map/reduce or requiring checkpoints to disk are 1000s of times too slow to extract the value latent in graphs for time-sensitive analytics. (2) Solutions based on non-updatable data representations are limited in their application to complex analytics. 3) Solutions that provide robust scaling and high performance require specialized programming techniques that are not easily accessible to the existing graph development community. Approaches leveraging multi-core technology have significant promise. At the purely hardware level, GPU memory bandwidth is set to jump by 4x by Q1 2016 (Pascal). This should provide a 4x speedup. Thus going from 10x - 100x speedups over CPUs to 40x - 400x over CPUs. PHASE I: Develop innovative approaches to apply many-core GPU and/or hybrid CPU technologies to existing graph development APIs. The focus should be on framework fidelity, computational scalability, and easing the burden of integration. In addition, develop detailed analysis of predicted performance of the proposed approach and plans for developing the approach into a comprehensive platform to accelerate a graph framework in Phase II. The Phase I deliverable is a final report documenting the effort and results. PHASE II: Develop a comprehensive implementation of an existing graph framework accelerated for commodity high performance many-core (GPUs) and multi-core CPUs technologies using the approaches identified in Phase I. Develop a prototype and establish a preliminary benchmark using various standard problems, and apply the tool to a DoD relevant problem. Phase II deliverables will include software, a final report documenting the effort, a document describing the architecture and a user's manual. PHASE III: Real time data ingest and reasoning analytics for military situational awareness platforms. Commercial uses of the accelerated graph framework include a 1000-10000X acceleration of existing graph analytics such as Facebook's current graph traversal.